UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/923,175	08/06/2001	Bruce Alan Vessey	TN134	1709

7590 08/11/2006

MICHAEL B. Atlass, Esq.
Unisys Corporation
Unisys Way, MS/E8-114
Blue Bell, PA 19424-0001

EXAMINER	
ANYA, CHARLES E	

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 08/11/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/923,175	VESSEY ET AL.	
	Examiner	Art Unit	
	Charles E. Anya	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE ____ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) _____ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☐ Claim(s) _____ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 65-100 are pending in this application.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 66 and 76 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

a. The following terms lack antecedent basis:

Claims 66 and 76 recite the limitation "the connection" in line 1 respectively. There is insufficient antecedent basis for this limitation in the claim.

For the purpose of this office action the examiner would change the phrase "the connection" to "the network connection".

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 65-68 and 75-78 are rejected under 35 U.S.C. 103(a) as b ing unpatentable over U.S. Pat. 5,652,885 to Reed et al. in view of U.S. Pat. No. 5,991,820 to Dean.

6. As to claim 65, Reed teaches a method for communicating between applications executing in different partition of a partitionable computer system (Col. 5 Ln. 12 – 15) the method comprising: receiving a request made by a first application on a first partition to establish a network connection with a second application on a second partition and to send a message to second application via the network connection (Setup Col. 8 Ln. 33 – 67): establishing a connection between the first partition and the second partition of the computer system through a memory region of the computer system shared by both the first partition and the second partition, wherein the connection emulates the requested network connection (Col. 5 Ln. 54 – 67, Col. 6 Ln. 1 – 11) and sending the message to the second application via the connection established through the shared memory region, whereby the connection established through the shared memory region appears to the first and second applications as the requested network connection (Col. 5 Ln. 54 – 67, Col. 6 Ln. 1 – 11); wherein the steps of establishing and sending comprise: creating a data structure in the shared memory region comprising a plurality of data segments forming a circular buffer (figures 4/5 Col. 22 – 65), writing, from the first partition on behalf of the first application, the message to one or more data segments, as needed (Col. 5 Ln. 57 – 65), and updating an indication of the data

segment containing the most recently written portion of the message (Col. 7 Ln. 66 – 67, Col. 8 Ln. 1 – 23, Step 712 Col. 10 Ln. 26 – 29).

Although Reed does not explicitly teach a first application on a first partition and second application on a second partition, the source user process and the destination user process would inherently be executed on separate partitions since one of the processes could be executed on a remote computer.

However, Dean a first application on a first partition and second application on a second partition (Col. 7 Ln. 45 – 50).

It would have been obvious to one of ordinary skill the art at the time the invention was made to combine the teachings of Dean and Reed because the teaching of Dean would improve the system of Reed by restricting memory access by processes (Col. 7 Ln. 45 – 50).

7. As to claim 66, Reed teaches the method recited in claim 65, wherein the connection requested by the first application comprises a socket connection (Col. 8 Ln. 54 – 66), and wherein the step of establishing a connection through the shared memory region comprises establishing a connection through the shared memory region that emulates a socket connection (Col. 5 Ln. 54 – 67, Col. 6 Ln. 1 – 11).

8. As to claim 67, Reed teaches the method in claim 65, further comprising: reading from the second partition on behalf of the second application the message from said one or more data segments and updating an indication of which data segments have

been read from the data structure (Col. 6 Ln. 51 – 62) and providing the message read from the data structure to the second application in accordance with an API associated with the requested network connection (Col. 6 Ln. 63 – 67, Col. 7 Ln. 1 – 21).

9. As to claim 68, Reed teaches the method of claim 65, wherein updating an indication of the data segment containing the most recently written portion of the message comprises incrementing the head index (Step 712 Col. 10 Ln. 26 – 29).

10. As to claims 75-78, see the rejection of claims 65 – 68 respectively.

11. Claims 69,70,71,74,79,80,81,84 and 96-100 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. 5,652,885 to Reed et al. in view of U.S. Pat. No. 5,991,820 to Dean as applied to claim 65 above, and further in view of U.S. Pat. No. 5,276,896 to Rimmer et al.

12. As to claim 69, Reed and Dean as modified are silent with reference to the method of claim 68, wherein updating an indication of which data segments have been read from the data structure comprising incrementing a tail index.

Rimmer teaches the method of claim 5, wherein updating an indication of which data segments have been read from the data structure comprising incrementing a tail index (Col. 16 Ln. 27 – 31).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Rimmer and Reed because the teaching of Rimmer would improve the system of Reed by providing a means for reading outstanding sent messages (Rimmer Col. 16 Ln. 27 – 31).

13. As to claim 70, Reed and Dean as modified are silent with reference to the method of claim 69, further comprising polling, by the receiving partition the shared memory region to determine if the message has been written to the shared memory region.

Rimmer teaches the method of claim 69, further comprising polling, by the receiving partition the shared memory region to determine if the message has been written to the shared memory region (figure 2b Col. 9 Ln. 43 – 57).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Rimmer and Reed because the teaching of Rimmer would improve the system of Reed by providing a means for determining when to send a message (Rimmer Col. 9 Ln. 43 – 57).

14. As to claim 71, Reed teaches the method of claim 70, further comprising receiving, by the receiving partition, an interrupt initiated by the sending partition and indicating that the message has been written to the shared memory region (Col. 6 Ln. 25 – 40).

15. As to claim 74, Reed and Dean as modified teach the method of claim 65, further comprising: an output queue for a given partition indicating whether that partition has placed in the shared memory region a message intended for any of the other partitions and if so, identifying a buffer containing the message (figure 3 Col. 6 Ln. 25 – 39), receiving at the first partition of the first application, said request to send a message to the second application via the requested type of network connection 9Col. 5 Ln. 13 – 22), writing, in response to the received request, the message to an available buffer in the shared memory region (Col. 5 Ln. 57 – 67) and indicating in the output queue of the first partition that the message has been written thereto (Col. 6 Ln. 9 – 11), determining at the second partition, from the output queue of the first partition, that the message has been placed in the buffer and retrieving the message from the buffer, and providing the message read from the data structure to the second application in accordance with an API associated with the requested network connection (Col. 6 Ln. 41 – 62).

Reed and Dean as modified are silent with reference to creating, in the shared memory region a plurality of output queues one for each of said first and second partitions, each partition polling the output queues of the other partitions to determine whether those other partitions have placed any messages intended for it in the shared memory region.

Rimmer teaches creating, in the shared memory region a plurality of output queues one for each of said first and second partitions (figure 1Col. 4 Ln. 1 – 39) and each partition polling the output queues of the other partitions to determine whether

those other partitions have placed any messages intended for it in the shared memory region (figure 2b Col. 9 Ln. 43 – 57).

16. As to claim 79, see the rejection of claim 69.

17. As to claims 80 and 96, see the rejection of claim 70.

18. As to claim 81, see the rejection of claim 71.

19. As to claims 84 and 97, see the rejection of claim 74.

20. As to claim 85, Reed teaches computer system comprising: a plurality of processing modules, groups of one or more processing modules being configured as separate partitions within the computer system, each partition operating under the control of a separate operating system (Source User Process 204, Destination User Process 302 Col. 5 Ln. 12 – 31, Col. 6 Ln. 25 – 40): a main memory to which each processing module is connected, the main memory having defined therein at least one shared memory region to which at least two different ones of said separate partitions have shared access (Col. 7 Ln. 29 – 65), and program code, executing on each of at least a first partition and a second partition of the computer system, wherein the program code establishes a connection between a first application on the first partition of the computer system and a second application on the second partition of the

computer system through the shared memory region, wherein the connection through the shared memory region emulates a network connection requested by one of the applications (IPC Mechanism 206 Col. 5 Ln. 12 – 41); wherein said program code executing on each of said first and second partitions comprises a shared memory service provider that serves as an interface between a component of the computer system that provides an API through which said application can make said request for a network connection and the shared memory region of the main memory through which the emulated network connection is established (Memory Allocation Subsystem 208 Col. 5 Ln. 32 – 41); wherein the shard memory service provider on each of said first and second partitions establishes a data structure in the shared memory region through which data is transferred from that partition to the shared memory (Memory Allocation Subsystem 208 Col. 5 Ln. 32 – 43); wherein the data structure comprises: a plurality of data segments, each of the plurality of data segments for storing network message data to be sent from a sending shared memory service provider to a receiving shared memory service provider (Col. 7 Ln. 22 – 65), a control segment for controlling reading and writing of data in the plurality of data segments, the control segment comprising: a first portion comprising: a first field for storing an indication of the data segment containing the most recently written network message data ("...block index..." Col. 5 Ln. 50 – 65).

Although Reed does not explicitly teach processing modules being configured as separate partitions, the source user process and the destination user process would

inherently be executed on separate partition since one of the processes could be executed on a remote computer.

However, Dean teaches processing modules being configured as separate partitions (Col. 7 Ln. 45 – 50).

It would have been obvious to one of ordinary skill the art at the time the invention was made to combine the teachings of Dean and Reed because the teaching of Dean would improve the system of Reed by restricting memory access by processes (Col. 7 Ln. 45 – 50).

Reed and Dean as modified are silent with reference to a second field for storing an indication of the data segment containing the earliest written, but not read, network message data, and a plurality of second portions, each second portion corresponding to one of the plurality of data segments for control of the data segment, each second portion comprising: a first field for storing an indication of the beginning of network message data within the data segment, and a second field fur storing are indication of the end of network message data within the data segment.

Rimmer teaches a second field for storing an indication of the data segment containing the earliest written, but not read, network message data (Col. 16 Ln. 27 – 31), and a plurality of second portions, each second portion corresponding to one of the plurality of data segments for control of the data segment each second portion comprising: a first field for storing an indication of the beginning of network message data within the data segment ("...TAKEPTR/PUTPTR..." Col. 10 Ln. 1 – 23, Col. 13 Ln. 9 – 21), and a second field for storing are indication of the end of network message data

within the data segment (Although neither Reed as modified nor Rimmer explicitly teaches this limitation official notice would be taken because it is well known in the art that a frame or message sent and received between application in a network inherently includes an End of Text (ETX) that indicates the end of the transmitted message).

21. As to claim 86, Rimmer teaches the computer system recited in claim 85, wherein the first portion further comprises: a third field for storing, an indication that the sending shared memory service provider is waiting to send the network message (Col. 14 Ln. 52 – 62) and a fourth field for storing an indication that the receiving shared memory service provider is waiting to receive the network message (Col. 11 Ln. 31 – 41).

22. As to claim 87, Reed and Dean as modified teach the computer system recited in claim 85, wherein each second portion further comprises a third field for storing an indication of a length of network message data within the data segment (Reed "...message length..." Col. 5 Ln. 15 – 19).

23. As to claim 88, Rimmer teaches the computer system recited in claim 87, wherein the plurality of data segments are linked to form a circular buffer (Col. 3 Ln. 1 – 3, Col. 4 Ln. 1 – 9), wherein each second portion further comprises: a fourth field for storing an indication of the next data segment in the circular buffer ("...next index..." Col. 10 Ln. 23 – 25), and a fifth field for storing an indication that the data segments

contains a last portion of a network message stored across a plurality of data segments ("...EOL..." Col. 16 Ln. 12 – 19).

24. As to claim 89, Reed teaches the computer system recited in claim 85, wherein the computer system provides a resource through which the shared memory service provider can establish the data structure and control the transfer of data through it, the resource providing the ability to perform at least one of the following operations on the shared memory region: (i) allocate an area of the shared memory region, (ii) map and unmap an area of the shared memory region, deallocate an area of the shared memory region, (iii) send and receive signals to and from other partitions via the shared memory region, and (iv) receive status information about the shared memory region and about selected partitions (Col. 5 Ln. 26 – 31).

25. As to claim 98, Rimmer teaches the computer system recited in claim 97, wherein for any communications placed in the shared memory region by the sending partition and intended to be received by another partition, the output queue of the sending partition specifies the location within the shared memory region of the a buffer containing that communication ("...TAKEPTR..." Col. 10 Ln. 1 – 20).

26. As to claim 99, Rimmer teaches the computer system recited in claim 98, wherein the program code executing on each of said first and second partitions further comprises a shared memory driver that receives a request to send a message to an

application on another partition, the request having been made in accordance with the application programming interface (API) associated with the requested type of network connection, and that, in response to the request, causes the message to be placed in an available buffer in the shared memory region and causes an indication of the message to be placed in the output queue of the sending partition (figure 5 Col. 13 Ln. 9 – 21).

27. As to claim 100, Rimmer teaches the computer system recited in claim 99, wherein the shared memory driver on each partition implements a same interface as network device driver to enable application programs and the operating system on that partition to send communications to other partitions via the shared memory region in the same manner that communications are sent to other computer systems over a network interface card (figure 2a Col. 9 Ln. 22 – 42, figure 5 Col. 13 Ln. 9 – 21).

28. Claims 72,73,82,83 and 90-95 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. 5,652,885 to Reed et al. in view of U.S. Pat. No. 5,991,820 to Dean as applied to claims 66,76 or 85 above, and further in view of U.S. Pat. No. 5,276,896 to Rimmer et al., and further in view of U.S. Pat. No. 5,825,890 to Elgamal et al.

29. As to claim 72, Reed and Dean as modified are silent with reference to the step of establishing a connection between the first partition and the second partition of the

computer system that emulates a socket connection further comprises performing the following steps on the second partition: (a) creating a socket on behalf of the second application to listen for attempts to connect thereto, (b) receiving a connect message from the first partition that identifies a memory location of the shared memory region at which the first partition has allocated a first data area to serve as a buffer for transferring data from the first partition to the second partition, (c) matching the received connect message to the listening socket created in step (a), (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the second partition to the first partition, (e) mapping both the first and second data areas into a process space of the listening socket, (f) initializing the second data area and (g) returning a connected indication to the first partition and informing the application on the second partition that the socket connection has been established.

Elgamal teaches the step of establishing a connection between the first partition and the second partition of the computer system that emulates a socket connection further comprises performing the following steps on the second partition: (a) creating a socket on behalf of the second application to listen for attempts to connect thereto (Col. 5 Ln. 55 – 58), (b) receiving a connect message from the first partition that identifies a memory location of the shared memory region at which the first partition has allocated a first data area to serve as a buffer for transferring data from the first partition to the second partition (Col. 5 Ln. 55 – 59), (c) matching the received connect message to the listening socket created in step (a) (Col. 5 Ln. 55 – 67), and (g) returning a connected

indication to the first partition and informing the application on the second partition that the socket connection has been established (Col. 12 Ln. 41 – 48).

It would have been obvious to one of ordinary skill the art at the time the invention was made to combine the teachings of Elgamal and Reed and Dean as modified because the teaching of Elgamal would improve the system of Reed and Dean as modified by providing means for receiving connection requests (Elgamal Col. 5 Ln. 55 – 61).

Reed, Dean and Elgamal as modified are silent with reference to (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the second partition to the first partition, (e) mapping both the first and second data areas into a process space of the listening socket, and (f) initializing the second data area.

Rimmer teaches (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the second partition to the first partition (Col. 11 Ln. 42 – 50), (e) mapping both the first and second data areas into a process space of the listening socket/(f) initializing the second data area (Col. 11 Ln. 51 – 56).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Rimmer and Reed, Dean and Elgamal as modified because the teaching of Rimmer would improve the system of Reed, Dean and Elgamal as modified by providing shared memory for coordinating message transfer (Rimmer Col. 10 Ln. 1 – 4).

30. As to claim 73, Reed teaches the method of claim 72, further comprising performing the following steps on the first partition: (a') receiving the request from the first application to establish the socket connection with the second application (Col. 8 Ln. 33 – 67), (b') creating a connecting socket (Step 606 Col. 8 Ln. 59 – 67), (c') allocating the first data area in the shared memory region (Col. 8 Ln. 46 – 54), (d') sending the connect message to the second partition that identifies the memory location of the shared memory region at which the first data area has been allocated (Col. 9 Ln. 5 – 23), and (e') upon receipt of the connected indication from the second partition, mapping the first and second data areas into a process space of the connecting socket to establish the socket connection between the first and second partitions ("...bind.../...attaches..." Col. 8 Ln. 57 – 67).

31. As to claims 82 and 83, see the rejection of claims 72 and 73 respectively.

32. As to claim 90, Elgamal teaches the computer system recited in claim 85, wherein the shared memory provider comprises: a dynamically link library (DLL) the executes in a user mode of the operating system of its respective partition, there by being an instance of the shared memory service provider DLL in a process space of each application in the partition the may request the establishment of a network connection (Col. 12 Ln. 22 – 29).

Although both Reed, Dean and Elgamal as modified are silent with reference to a device driver the executes in a kernel mode of the operating system of the respective

partition, there being on one instance of the device driver in each partition, official notice is taken because this limitation is well known in the art.

33. As to claim 91, Elgamal teaches the computer system recited in claim 85, wherein the connection established through the shared memory region emulated a socket connection (Col. 5 Ln. 55 – 67).

34. As to claim 92, Reed, Dean and Elgamal as modified teach the computer system recited in claim 91, wherein said program code executing on each of the said at least first and second partitions comprises a shared memory service provider that serves as an interface between a component of the computer system that provides an API through which an application can make a request for a socket connection and the shared memory region of the main memory through which the emulated socket connection is established (Reed Col. 8 Ln. 57 – 67).

35. As to claim 93, Elgamal teaches the computer system recited in claim 92, wherein the operating system in each partition comprises a Microsoft Windows operating system, and wherein the component of the computer system that provides the API of the requested socket connection comprises a Winsock DLL and a Winsock Switch, the Winsock DLL forwarding a request for a socket connection made by an application in a given partition to the Winsock Switch, wherein the Winsock Switch allows multiple service providers, each of which provide TCP/IP services, to service

such a request, and wherein the shared service provider acts as a TCP/IP service provider so that a request for an application for socket connection can be serviced by the shared memory service provider (Elgamal Col. 5 Ln. 15 – 54).

36. As to claim 94, Reed, Dean and Elgamal as modified teach the computer system recited in claim 93, wherein the shared memory service provider on a first partition that represents the listening side of a requested socket connection performs the following steps: (a) creating a socket on behalf of a first application executing in the first partition in order to listen for attempts to connect thereto (Reed figure 6 Col. 8 Ln. 57 – 67), (c) matching the received connect message to the listening socket created in step (a) (Reed Col. 59 – 67).

Reed, Dean and Elgamal as modified are silent with reference to (b) receiving a connect message from the shared memory service provider on the second partition that identifies a memory location of the shared memory region at which the shared memory service provider on the second partition has allocated a first data area to serve as a buffer for transferring data from the second partition to the shared memory service provider on the first partition, (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the first partition to the shared memory service provider on the second partition, (e) mapping both the first and second data areas into a process space of the listening socket, (f) initializing the second data area and (g) returning a connected indication to the shared memory service provider on

the second partition and informing the application on the first partition that a socket connection has been established.

Rimmer teaches (b) receiving a connect message from the shared memory service provider on the second partition that identifies a memory location of the shared memory region at which the shared memory service provider on the second partition has allocated a first data area to serve as a buffer for transferring data from the second partition to the shared memory service provider on the first partition (Col. 10 Ln. 30 43), (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the first partition to the shared memory service provider on the second partition (Response Ring Buffer 184 Col. 10 Ln. 1 – 6, Col. 11 Ln. 42- 46), (e) mapping both the first and second data areas into a process space of the listening socket/(f) initializing the second data area (Col. 11 Ln. 42 – 50) and (g) returning a connected indication to the shared memory service provider on the second partition and informing the application on the first partition that a socket connection has been established (Col. 11 Ln. 42 – 50).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Rimmer and Reed because the teaching of Rimmer would improve the system of Reed by providing shared memory for coordinating message transfer (Rimmer Col. 10 Ln. 1 – 4).

37. As to claim 95, Elgamal teaches the computer system recited in claim 94, wherein the shared memory service provider on the second partition performs the

following steps: (a') receiving a request from an application on the second partition to establish a socket connection with the first application on the first partition (Col. 5 Ln. 40 Ln. 40 –43), (b') creating a connecting socket on the second partition (Col. 5 Ln. 55 – 58) and (c') allocating the first data area in the shared memory region (Col. 5 Ln. 58 – 59)

Reed as modified and Elgamal are silent with reference to (d') sending the connect message to the first partition that identifies the memory location of the shared memory region at which the first data area has been allocated and (e') upon receipt of the connected indication form the first partition, mapping the first and second data areas into a process space of the connecting socket to establish the socket connection between the first and second partitions.

Rimmer teaches (d') sending the connect message to the first partition that identifies the memory location of the shared memory region at which the first data area has been allocated (Col. 11 Ln. 42 – 50) and (e') upon receipt of the connected indication form the first partition, mapping the first and second data areas into a process space of the connecting socket to establish the socket connection between the first and second partitions (Col. 11 Ln. 51 – 56).

Response to Arguments

38. Applicant's arguments filed 11/16/05 have been fully considered but they are not persuasive.

Applicant argues in substance that (1) the Reed and Dean prior arts does not teach "network connection" and (2) the Reed and Dean prior arts merely teaches only a single operating system and such does not disclose different partitions/applications operating under the control of separate operating system.

Examiner respectfully traverses Applicant's argument:

As to point (1), claim 2 of the instant application indicates the network connection requested by the first application is a socket connection. Figure 5 (column 7 lines 27 – 52) of the Dean's prior art describes how to request and establish network connection via a socket system call, thus negating Applicant's assertion that neither Reed or Dean prior art teaches the "the network connection" recited in the claims.

As to point (2), contrary to Applicant's argument the Dean's prior art discloses the method of operating processes/programs in distributed window system. This method further includes operating the processes/programs on different computer systems, the different computer systems includes **different computer architecture and different window systems** (Column 5 lines 1 – 12). The different computer architecture and different window systems as describe here implies that the distributed window system operates the processes/programs under the control of different/separate operating system.

Conclusion

39. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Charles E Anya whose telephone number is (703) 305-3411. The examiner can normally be reached on M-F (8:30-6:00) First Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-Ai can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Charles E Anya

Examiner
Art Unit 2126

cea.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100